REMARKS/ARGUMENTS

The Applicant acknowledges the receipt of the Office Action mailed June 1, 2004. Claims 1-32 are now pending and claims 1-22 stand rejected. Reconsideration and allowance of claims 1-22, as amended, is respectfully requested. In addition, Applicant has added claims 23-32. Accordingly, amended claims, new claims, and supporting remarks are hereby presented that particularly point out and distinctly claim the subject matter that Applicant regards as his invention. No new matter is being added.

I. Rejection of claims 1, 8 and 21-22 under 35 U.S.C. 103(a)

Independent claims 1 and 8 stand rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Buser et al. (U.S. Pat. No. 6,507,921) in view of Zolnowsky et al. (U.S. Pat. No. 4,566,063).

Buser et al. is directed to a method of operating a digital system including a microprocessor operable to trace a sequence of instruction addresses. The method is particularly useful in optimizing execution performance and may also be useful in debugging program code. Buser et al. discloses "repeat( )" and "blockrepeat( )" instructions in the cited portion at col. 22, lines 36-41 and addresses the traceability of addresses associated therewith. The debugger has the option of unrolling all the information inside the repeat blocks; see col. 22, lines 53-54. Thus, Buser et al. teaches an unrolling method as described in Applicants' application as prior art, see page 7, line 20 to page 8, line 2 describing Figure 1B. To summarize, an unrolling method provides an effective data rate of one transfer <u>every two</u> clock cycles.

Zolnowsky et al. is directed to a data processor that can repeat the execution of instruction loops with minimal instruction fetches. More specifically, Zolnowsky et al. teaches a "pipelined" system. In a pipelined system, a number of instructions are fetched and used to fill an instruction stream that is then executed. The instruction stream is rebuilt, e.g., refetched, each time the instructions are repeated. This is, in essence, a looping system. Applicants' show a looping method as prior art in Figure 1A, the description of which is found at page 7, lines 9-19. A looping method provides an effective data rate of one transfer every three clock cycles.

Independent claims 1 and 8 have been amended to more particularly point out and distinctly claim the subject matter that Applicant regards as his invention. Claim 1 as now amended recites that the instruction is repeatedly executed a consecutive number of times as indicated by the count value. Similarly, claim 8 as now amended recites a means for repeatedly executing the instruction a consecutive number of times as indicated by the count value. Thus, the present invention thereby provides an effective data rate of one transfer every clock cycle. Buser et al. and Zolnowsky et al. fail to disclose or suggest repeatedly executing an instruction a consecutive number of times as indicated by a count value.

Claims 21 and 22 are dependent claims, depending from claim 8 and therefore contain each and every element of claim 8. Therefore, for the reasons already set forth for claim 8, claims 21 and 22 are also allowable. Reconsideration and allowance of claims 1 and 8, as well as claims 21 and 22 which depend from claim 8, are therefore respectfully requested.

II. Rejection of claims 2-7 under 35 U.S.C. 103(a)

Independent claims 2 and 3 stand rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Buser et al. in view of Shridhar et al. (U.S. Pat. No. 5,727,194). As previously described, Buser et al. teaches an unrolling method in which an effective data rate of one transfer every two clock cycles is provided.

Shridhar et al. is directed to a repeat-bit based system in which loops are repeatedly executed. Shridhar et al., like Zolnowsky et al., teaches a pipelined system in which sequence of instructions is repeatedly fetched, decoded, and executed. According to the teachings of Shridhar et al., upon detection of a repeat instruction, the first loop instruction is refeteched.

(col. 14, lines 1-5) In other words, Shirdhar et al. teaches a looping function. Again, a looping method provides an effective data rate of one transfer every three clock cycles.

Therefore, the combination of Buser et al. and Shridhar et al. also does not result in Applicant's invention as claimed. However, Applicant has amended claims 2 and 3 to more particularly point out and distinctly claim the subject matter that Applicant regards as his invention. Claim 2 as now amended recites that the instruction is repeatedly executed a consecutive number of times as indicated by the count value. Claim 3 as now amended implicitly implies that the single instruction is repeatedly executed a consecutive number of times as indicated by the count value, and by decrementing the count value. Thus, the present invention thereby provides an effective data rate of one transfer every clock cycle. Buser et al. and Shridhar et al. fail to disclose or suggest repeatedly executing an instruction a consecutive number of times as indicated by a count value, implicitly or otherwise.

Claims 4-7 are dependent claims, depending from claim 3 and therefore contain each and every element of claim 3. Therefore, for the reasons already set forth for claim 3, claims 4-7 are also allowable. Reconsideration and allowance of claims 2 and 3, as well as claims 4-7 which depended from claim 3, are therefore respectfully requested.

III. Rejection of claims 9-20 under 35 U.S.C. 103(a)

Independent claims 9, 10, and 15 stand rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over Buser et al. in view of Kiuchi et al. (U.S. Pat. No. 5,579,493). Again, Buser et al. teaches an unrolling method in which an effective data rate of one transfer every two clock cycles is achieved.

Kiuchi et al. is directed to a system with a loop buffer and a repeat control circuit having a stack for storing control information. The loop buffer is used for storing instructions that are repeated executed in order to decrease power consumed by a processor. More specifically, an instruction to be repeated is read out of memory and written into a buffer. The buffer also holds a number of instructions to be looped. When the instructions are repeated, i.e., looped, the instructions are refetched from the buffer rather than from the memory. Thus, Kiuchi reduces bus traffic. See col. 8, lines 21-29 describing Figure 1. In other words, Kiuchi et al., likes Shirdhar et al., teaches a looping method that refetches instructions. A looping method, again, provides an effective data rate of one transfer every three clock cycles.

Therefore, the combination of Buser et al. and Kiuchi et al. also does not result in Applicant's invention as claimed. However, Applicant has amended claims 9, 10, and 15 to more particularly point out and distinctly claim the subject matter that Applicant regard as his invention.

The present invention generally includes a register for storing a count indicative of the number of times a single instruction will be repeatedly executed and a program counter. The count stored in the register is decremented each time the instruction is executed. Once the count value is less than or equal to zero, the program counter is incremented. Buser et al. and Kiuchi et al. fail to disclose or suggest decrementing a count value in a register each time a single instruction is executed and incrementing a program counter once the count value in the register is less than or equal to zero.

Applicant has now amended claims 9 and 10 to include the limitations of decrementing a count value in a register each time the single instruction is executed and incrementing a program counter once the count value in the register is less than zero and equal to zero, respectively. Thus, similar to claim 3, claim 9 implicitly implies that the single instruction is repeatedly executed a consecutive number of times as indicated by the count value, and by decrementing the count value. Applicant has also now amended claim 10 to indicate that the instruction is repeatedly executed a consecutive number of times as indicated by a count value. Likewise, Applicant has amended claim 15 to include the limitations of storing a count in the general purpose register and decrementing the count stored in the general purpose register each time the single instruction is executed and incrementing the program counter once the count stored in the general purpose register is below a threshold value using a state machine. Applicants have also amended claims 9, 10, and 15 to point out that an effective data rate of one transfer every clock cycle is provided. Buser et al. and Kiuchi et al. fail to disclose or suggest repeatedly executing an instruction a consecutive number of times as indicated by a count value, implicitly or otherwise.

Claims 11-14 are dependent claims, depending from claim 10 and therefore contain each and every element of claim 10. Therefore, for reasons already set forth for claim 10, claims 11-14 are also allowable. Similarly, claims 16-20 are dependent claims, depending from claim 15 and therefore contain each and every element of claim 15. Therefore, for the reasons set forth for claim 15, claims 16-20 are also allowable. Reconsideration and allowance of claims 9, 10, and 15, as well as claims 11-14 which depended from claim 10 and claims 16-20 which depended from claim 15, are therefore respectfully requested.

IV. New claims 23-32 are in condition for allowance under 35 U.S.C. 102 and 35 U.S.C. 103

Claims 23-25 are dependent upon claim 15 and therefore contain each and every element of claim 15. Therefore, for the reasons already set forth for claim 15, claims 23-25 should be in condition for allowance. Furthermore, claim 23 recites incrementing the program counter once the count value is zero, which is neither disclosed, taught, or motivated by the cited prior art. In addition, claim 24 recites incrementing the program counter once the count value is less than zero, which also is neither disclosed, taught, or motivated by the cited prior art. Likewise, claim 25 recites that the program counter remains unchanged as the single instruction is repeatedly executed, which the cited prior art does not teach, suggest, or show.

Claims 26-28 are dependent upon claim 8 and therefore contain each and every element of claim 8. Therefore, for the reasons already set forth for claim 8, claims 26-28 should be in condition for allowance. Furthermore, claim 26 recites incrementing a program counter once the count value is zero, which again is neither disclosed, taught, or motivated by the cited prior art. Claim 27 recites incrementing a program counter once the count value is less than zero, which also is neither disclosed, taught, or motivated by the cited prior art. Claim 28 recites that a program counter remains unchanged as the single instruction is repeatedly executed, which again the cited prior art also does not teach, suggest, or show.

Claims 29 and 30 are dependent upon claim 2 and therefore include each and every element of claim 2. Therefore, for the reasons already set forth for claim 2, claims 29 and 30 should be in condition for allowance. Additionally, claim 29 recites that the program counter remains unchanged as the single instruction is repeatedly executed, which the cited prior art does not teach, suggest, or show. Claim 30 further recites that the program counter is effectively stalled on the single instruction until the single instruction executes the number of times indicated by the count value, which the cited prior art also does not teach, suggest, or show.

Claims 31 and 32 are dependent upon claim 3 and therefore include each and every element of claim 3. Therefore, for the reasons already set forth for claim 3, claims 31 and 32 should be in condition for allowance. Additionally, claim 31 recites that the program counter remains unchanged as the single instruction is repeatedly executed, which the cited prior art does not teach, suggest, or show. Claim 32 further recites that the program counter is effectively stalled on the single instruction until the single instruction executes the number of times indicated by the count value, which the cited prior art also does not teach, suggest, or show.

In summary, Applicant respectfully submits that all pending claims are novel and non-obvious over the prior art of record. Reconsideration and allowance of all pending claims are therefore respectfully requested. If the Examiner believes there are any further matters that need to be discussed in order to expedite the prosecution of the present application, the Examiner is invited to contact the undersigned.

If there are any other fees necessitated by the foregoing communication, please charge such fees to our Deposit Account No. 50-0902, referencing our Docket No. (72255/02662).

Respectfully submitted,

TUCKER ELLIS & WEST LLP

Date: 8-27-04

Larry B. Donovan
Registration No. 47,230
1150 Huntington Building
925 Euclid Avenue
Cleveland, Ohio 44115-1475
Customer No. 23380
(216) 696-3864 (phone)
(216) 592-5009 (fax)

CERTIFICATE OF TRANSMISSION UNDER 37 C.F.R. §1.8

I hereby certify that this correspondence (along with any paper referenced as being attached or enclosed) is being deposited on the below date with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, P. O. Box 1450, Alexandria, VA 223143-1450, or being facsimile transmitted to the USPTO, on the date indicated below.

Date: 8-27-04

Name: Amy E. Gagich